

***Amendments to the Abstract***

Please amend the Abstract as follows:

~~A method of on~~On-the-fly patching of executable code includes placing a block of modified instructions in memory, identifying a block of code to be ~~changed~~ patched, storing instructions to be ~~changed~~ patched from the block of code to be ~~changed~~ patched in a storage location, change the instructions to be ~~changed~~ patched to mark instructions, and adding a jump to the block of modified instructions in the block of code ~~to be changed~~. Prior to the placing and the identifying steps, a write flag for a page in memory where the block of code to be ~~changed~~ patched is located is set to allow writes, and interrupts are masked. The instructions are replaced in reverse order. The mark instructions are the same length, in bytes, as the instructions to be ~~changed~~ patched. The modified instructions include a resolver to determine a number of instructions of the block of code to be ~~changed~~ patched that had already been executed. ~~If the number is less than a number of instructions to be changed, then a “no patch installed” scenario is imitated.~~